第 12 卷 第 29 期 2012 年 10 月
1671—1815(2012)29-7505-06

科 学 技 术 与 工 程
Science Technology and Engineering

Vol. 12 No. 29 Oct. 2012
© 2012 Sci. Tech. Engrg.

生物科学

基于统计分析与分段码书的 DNA 序列压缩新方法

熊文萍 孙季丰

(华南理工大学电子与信息学院,广州 510640)

摘 要 将 DNA 序列分成 64 个碱基一组的短序列。根据每个小段落不同的碱基排列特点,通过对每段中重复频率最高的三碱基组合片段采用特定码书编码,提出了基于统计分析与分段码书的 DNA 序列压缩方法,以达到对 DNA 数据压缩的目的。实验表明,本算法在大部分常用基准测试序列中达到了比较好的压缩性能。

关键词 DNA 序列 统计分析 码书 分段编码

中图法分类号 Q523.8; **文献标志码** A

生物基因技术和 DNA 测序是很重要的新兴科技领域,它包含了生物学、医学、遗传学、生物信息学等方面的理论和技术。随着人类基因组等基因工程的展开,DNA 数据正呈爆炸式的增长[1]。面对这种情况,如何有效地进行传输和存储也成了计算机专家和生物信息学家面临的新问题。对海量的 DNA 数据进行有效压缩,存储于较小的空间成为必然的解决方式[2]。DNA 数据不同于一般的文本数据或其他数据,有其独到的特点。首先它是仅由 A、G、C、T 四种字符以任意顺序排列而成的长字符串,其次由于 DNA 是生物学数据,所以序列中蕴含着特定的生物学特点,比如重复性。因此,传统的压缩算法并不能高效地压缩 DNA 序列,因而出现了一系列针对 DNA 序列的专用压缩算法。

DNA 序列是由四种碱基:腺嘌呤(A)、胸腺嘧啶(T)、鸟嘌呤(G)和胞嘧啶(C)线性排列而成的碱基序列。其数据压缩算法可以分为两大类,一种是基于替代的方法,另一种是基于统计的方法。Grumbach 和 Tahi 提出了 Biocompress 算法[3],该算法在 LZ 算法的基础上加入了互补回文结构。Chen 和 Kwong 提出了 Gencompress 算法[4],增加了对近似重复序列的编码处理。Matsumoto 和 Sadakane 提出了 CTW + LZ 算法,使用了多个模型对 DNA 序列的不同片段进行压缩。X Chen 和 M Li 提出了 DNAcompress 算法[5],算法中引入 PatternHunter 工具搜索 DNA 序列的重复与近似重复片段,提高了算法速度。纪震,周家锐等提出了 BioLZMA 算法,将序列的生物学特征运用到了编码中[6]。

以上方法都考虑到了 DNA 序列作为生物学数据的特点,对序列中的精确重复和近似重复序列进行了编码,达到了比较好的压缩效果。但是以上方法都是针对序列中的长重复片段或长近似重复片段,而序列中重复频率更高的短片段却被忽略了。现将 DNA 序列进行分段考虑,然后充分利用短片段的高度重复性,对每段中重复次数最多的三碱基组合进行编码,达到了比传统方法更好的压缩率。

1 DNA 数据统计特性分析

在生物 DNA 序列中,基因只是其中很小的一部分片段,而在基因组中,编码序列也只占其中很小一个比例。DNA 序列可以分为两大类:一类是单一序列,表示在整个 DNA 序列中,这种核苷酸的排列次序只出现过一次或只有一个拷贝,另一类是重复

2012 年 6 月 11 日收到,6 月 26 日修改

第一作者简介:熊文萍,女。华南理工大学电子与信息学院硕士研究生。研究方向:信号与信息处理。E-mail:642384418@qq.com。

序列,表示某种碱基的排列次序在整个 DNA 序列中出现的次数或其拷贝数有多次。构成基因的大多是单一序列,但是对于前面提到的那些非编码区域,则大部分由重复序列组成。重复序列根据重复的次数不同又可以分为轻度重复、中度重复和高度重复序列[7]。传统的 DNA 序列数据压缩方法主要考虑了重复度相对较低的长 DNA 序列,以及近似重复的序列,而对高度重复的短 DNA 序列片段则考虑较少。

重复序列分为串联重复序列和分散重复序列,串联重复序列表示不同数目的碱基组合重复拷贝串联在一起的高度重复序列,例如卫星 DNA 就是典型的串联重复序列。分散重复序列是指分布于 DNA 序列中散在的重复序列,一般为中度重复序列[7]。DNA 序列数据中可以利用重复序列的串联性和散在性来进行高效的压缩。

如图(图 1—图 3)所示给出了 DNA 数据(CHMPXX)中,不同的五碱基组合片段,四碱基组合片段和三碱基组合片段的重复情况,图中横坐标为碱基组合对应的十进制数,对于三碱基组合,横坐标为 $0\sim4^3-1$,四碱基组合 $0\sim4^4-1$,五碱基组合为 $0\sim4^5-1$。纵坐标表示不同碱基组合在 DNA 数据中重复的总次数。由图可知,同碱基数目但不同排列的碱基组合在序列中重复的概率有很大差异。

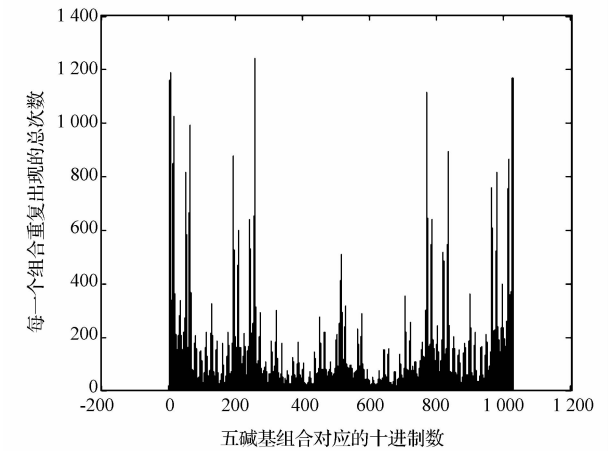

图 1 五碱基组合重复情况

表 1 统计出了不同 DNA 序列数据中,重复概率最大的三碱基组合、四碱基组合以及五碱基组合这三种短序列以及其重复出现的次数。

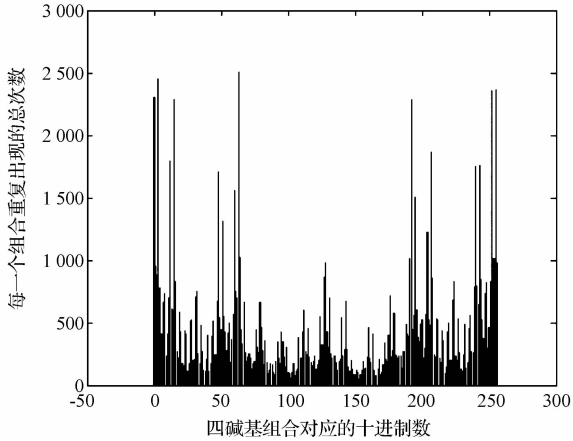

图 2 四碱基组合重复情况

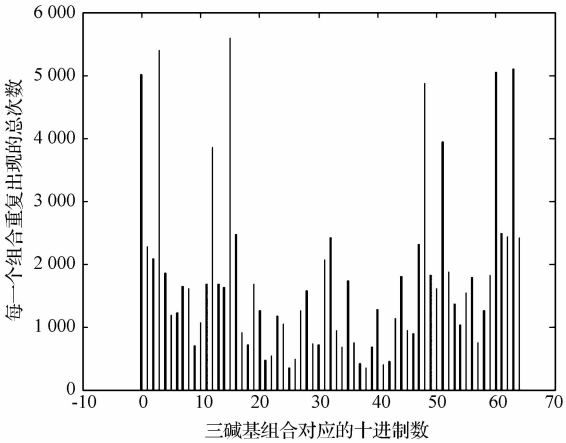

图 3 三碱基组合重复情况

以上统计结果表明,在整个 DNA 序列中,不同数目的碱基组合重复概率差异很大,长序列比短序列重复出现的概率小。例如,对于特定的基准序列 VACCG,其碱基总数为 194 711,三碱基组合中,ATA 出现概率最大,为 7 156 次,而四碱基组合中,ATAT 出现概率最大,为 2 607 次,五碱基组合中,AATAT 出现概率最大,为 1 044 次。由以上数据可以发现,序列越短,重复概率越大。而且序列每增加一位,重复的概率就会大大减小。可以推断,二碱基组合重复出现的概率更大,但是其对于实际序列压缩意义不大,所以不予考虑。从生物学意义分析,在重复度相对较低的基因编码区,如外显子部分,三个碱基一起可以决定一个密码子,有些密码子出现的

表 1　常见 DNA 数据短片段重复特性

DNA 序列	长度	三碱基组合	四碱基组合	五碱基组合
HUMGHCSA	66 495	CTG 1 726 次	CTGG 588 次	AGAAA 220 次
HUMHDABC	58 864	CTG 1 700 次	CCTG 598 次	CTGGG 241 次
CHMPXX	121 024	ATT 5 592 次	ATTT 2 502 次	ATTTT 1 239 次
HEHCMVCG	229 354	GCG 6 448 次	CGCG 2 146 次	CGGCG 930 次
VACCG	194 711	ATA 7 156 次	ATAT 2 607 次	AATAT 1 044 次

概率大,因此特定的三碱基组合也会以大概率出现。综合上述考虑,文章算法中考虑三碱基组合的重复情况。

算法中,并不是直接对整个 DNA 序列考虑某一种三碱基组合的重复情况,而是先将 DNA 序列分成 64 个碱基一组的短片段,再对每一段进行独立分析。每一段序列的碱基排列特点不同,统计每一段中重复出现频率最高的三碱基组合,然后对该碱基组合进行编码。因为文章算法统计的是三碱基组合的重复频率,若直接对每一个碱基用两位二进制编码,则三碱基组合所用的编码位数为 6 位,6 位二进制所能表示的最大十进制数为 63(从 0 算起),因此对 DNA 分段时,采用 64 个为一组的方法,以方便后续压缩编码。

3　压缩算法的实现

该压缩算法中,首先对 DNA 数据 s 进行预处理,然后将处理过的序列用分段码书编码,最后对完整 DNA 数据编码压缩。

在 DNA 数据的预处理阶段,将待压缩的 DNA 数据从第一个碱基开始,依次分成 64 个碱基一组的序列片段,对于最后不足 64 个碱基的先不考虑。

将预处理后的 DNA 数据片段利用分段码书进行编码。

Step1:初始化 $i=1$,$num_continue=0$,集合 $S=\{\}$,字符串 $code=''$,$code_al=''$,$code_no=''$。

Step2:统计第 i 个分组中出现频率最高的三碱基组合,每两个组合之间碱基无重合。用集合 S 记录该三碱基组合的编码相关信息:$S=\{pos,num,c,d\}$,其中 pos 表示三碱基组合第一次出现相对于该段起始点的位置,num 表示该三碱基组合在本段中重复出现的总次数,c 表示用多少位二进制来编码每相邻两次出现之间的间隔距离,$d=\{d_1,d_2,\cdots,d_n\}$ 表示相邻两次出现之间的间隔距离集合,n 表示距离的个数,其值比 num 小 1。

Step3:对该三碱基组合的信息集 $S=\{pos,num,c,d\}$,按照如下规则进行编码,并将编码好的二进制序列存入 $code$ 中。

(1)pos:表示该三碱基组合第一次出现相对于该段起始点的位置。用固定的 5 位二进制编码 pos,编码的范围为 0 ~ 31。若某段的三碱基组合第一次出现的位置大于 31,则直接放弃对本段的编码。

(2)num:表示本段中该三碱基组合出现的总次数。实验发现,小于 3 次的不利于最后的压缩,所以实际压缩中不考虑。因此出现次数从 3 计起,也即,若该段中任意三碱基组合出现的最高次数小于 3,则不对本段编码。实验统计表明,通常情况下,次数集中在 5 到 8 次,所以用固定的 3 位二进制编码次数 num。3 位二进制表示 num 值 3 到 9,如:用 000 表示 3 次,100 表示 7 次。若次数超过 9 次,则先存入 111,然后紧接着用 3 位二进制编码大于 9 的部分,如若次数 num 为 15,则 num 编码为 111110。

(3)c:表示三碱基组合每相邻两次出现之间的距离 d_i 需要用多少位二进制来编码,用 dis 来表示。用固定的两位二进制编码 c,用 00 表示 $dis=1$(编码距离为 0 或 1),用 01 表示 $dis=2$(编码距离为 0 ~ 3),用 10 表示 $dis=3$(编码距离为 0 ~ 7),用 11 表示 $dis=4$(编码距离为 0 ~ 15)。若相邻两次出现之间的距离大于最大值 15,则用 0000 给予标识后,接下来用固定 3 位二进制数编码大于 15 的部分。若此种方式仍不足于表示相邻距离,则直接将出现在本次以后的所有三碱基组合都舍弃不予考虑。

c 取值的确定,统计出三碱基组合每相邻两次出现之间的距离值集合 $d=\{d_1,d_2,\ldots,d_n\}$ 之后,根据集合中的最大值 $\max(d)$ 和次大值来确定所采用

的编码位数 dis。

a)若根据最大值和次大值确定的编码位数相同,则用该位数值来编码每一个距离值。例如,若最大距离值为 7,编码此距离需用 3 位二进制,次大距离值为 5,编码此距离也需用 3 位二进制。则本段编码取 $dis=3$,c 编码为 10。

b)若根据最大值和次大值确定的编码位数不同,则取次大距离作为确定编码位数的标准。例如,若最大距离为 7,编码此距离需用 3 位二进制,而次大距离为 3,则编码此距离只需用 2 位二进制。此时,$dis=2$,c 编码为 01。对于最大距离 7,用'00'给予标识后,用 3 位二进制编码大于 3 的部分。

(4)$d=\{d_1,d_2,..,d_n\}$:表示本段中三碱基组合每相邻两次出现之间的距离值集合,按 c 编码中约定的二进制位数对每一个距离值 d_i 编码。

a) 若 $d=\{d_1,d_2,..,d_n\}$ 中每相邻两个值互不相等,则直接将每一个值分别编码成固定位数的二进制。

b) 若 $d=\{d_1,d_2,..,d_n\}$ 中有连续几个值 d_k,$d_{k+1},..,d_{k+s}$ 相等,则用形如 $\{d_k,'11',s+1\}$ 的组合来编码。如:假设有距离值集合为 $\{3,4,4,4,4,4,6\}$,则编码为 $\{011,\{100,11,101\},110\}$。

Step4:计算第 i 段的效益函数值 G_i,用来判断第 i 段的编码是否有利于最终的压缩。其中效益函数 $G(num,code)$ 定义如下

$$G(num,code)=num\times 6-length(code) \quad (1)$$

式中 num 乘以 6 表示直接用两位二进制编码这些三碱基组合的每一位时所需要的二进制总位数,$length(code)$ 表示采用上述分段编码算法编码三碱基组合所需要的二进制总位数。用公式表示为

$$length(code)=length(pos)+length(num)+length(c)+\sum_{i}^{n}length(d_i) \quad (2)$$

公式可简化成

$$length(code)=5+3+2+disn+w=10+disn+w \quad (3)$$

式(3)中,w 表示编码过程中表示附加信息所用的编码位数。

Step5:根据不同的 G_i 值做不同的编码处理。

(1) 若 $G_i>0$,则根据 $num_continue$ 的取值进行编码。

a)若 $num_continue>1$, 用形如 $\{'00',num_continue\}$ 的方式标识第 i 段之前不参与分段码书编码的所有段落,并且 $num_continue=1$,转入 b),否则直接转入 b)。

b)将第 i 段的编码序列 $code$ 存入 $code_al$ 中,$code_al$ 表示已经编码的二进制序列。将本段中参与了编码的三碱基组合去掉后剩下的部分存入 $code_no$ 中,$code_no$ 表示未处理核苷酸序列。为了便于解码,去掉三碱基组合的时候不能去掉第一次出现的三联体。

(2)若 $G_i\leqslant 0$,根据 $i-1$ 段的 G 值 G_{i-1} 进行编码。

a)若 $G_{i-1}>0$,则在序列 $code_al$ 中存入'00',用于标识本段不参与分段码书编码,并且将本段序列完整地存入 $code_no$ 中,用于等待后续编码。

b)若 $G_{i-1}\leqslant 0$,变量 $num_continue=num_continue+1$ 来记录 $G<0$ 的连续片段个数,并且将本段序列完整地存入 $code_no$ 中。

Step6:若 $64i\leqslant length(s)$,且 $length(s)-64i\geqslant 64$,则 $i=i+1$,$G=0$,$code=''$,然后转入 Step2。

Step7:最后将 DNA 数据预处理时,分组最后不足 64 个碱基的序列段存入 $code_no$ 中。

至此,完成了对 DNA 数据的第一遍编码,得到两个序列,$code_al$ 和 $code_no$ 其中前者是已经编码好了的二进制 bit 流,后者是未处理序列,可以直接运用传统的方法(如 Gencompress 算法等)对本序列进行编码,最后达到对 DNA 序列压缩的目的。

举例说明对每个序列段的处理:

AGGTGCTACTGGCCTGTAGCTACTGGGGCTAG

ATAGCTGCGATCGGCTATCGTAGCTTGCTGTC

如上所示,该序列段总共有 64 个碱基,其中粗体标注的 GCT 为本段中出现次数最多的三碱基组合。

可以看出,GCT 总共出现了 7(num 值)次,第一次出现的位置相对于起始点为 5(pos 值),每相邻两次出现之间的距离值集合为:$\{11,6,5,7,6,1\}$($d=\{d_1,d_2,\cdots,d_n\}$ 值)

按照上面算法思路,根据最大距离值值 11 和次大距离值 7 确定距离编码所用位数,编码 11 需用 4 位二进制,编码 7 需用 3 位二进制,因此 $dis = 3$,c 编码为 10。对本段中三碱基组合的记录信息集合

$$\{pos, num, c, d = \{d_1, d_2, \cdots, d_n\}\}$$

编码的结果为:

{00101,100,10,000100,110,101,111,110,001}

编码总位数 $length(code) = 31$,则 $G(num, code) = num \times 6 - length(code) = 7 \times 6 - 31 = 11$。$G > 0$,将三碱基组合的 *code* 编码存入 *code_al* 序列中,将序列中第二个 GCT 及以后的 GCT 去掉后存入 *code_no* 序列中,以进行后续编码。

3　实验结果

实验中使用美国 Genbank 数据库中的标准 DNA 数据文件作为算法测试数据,将提出的算法分别作用于这些测试数据,并将性能结果与经典的 DNA 序列压缩算法进行比较,如 Biocompress 算法,Gencompress 算法,BioLZMA 算法等。实验中用压缩作为评价压缩性能的主要标准。压缩率表示为

$$ratio = 1 - \frac{length(code)}{length(s) \times 2} \tag{4}$$

式(4)中,*code* 表示编码后的二进制符号,*s* 为压缩所用的 DNA 数据。

测试用的基准序列包含了不同物种不同功能的 DNA 数据片段,能够有效评估压缩算法对含有不同数据特性的 DNA 序列的压缩能力。

算法压缩率如表 2 所示。

由本文算法与传统的压缩算法压缩率对比结果可以看出,在考虑了分段 DNA 序列中三个碱基组合的重复情况后,可以提高序列的压缩率。尤其是对于那些不同三碱基组合重复概率差异很大的 DNA 序列数据,本压缩算法的压缩率优势更明显。

4　总结

将 DNA 数据进行了分段预处理,划分成 64 个碱基一组的短片段,然后根据每一段的统计学特征,分别进行编码压缩处理。算法中充分利用了 DNA 数据三碱基组合的高度重复性,有效提高了压缩效率。实验表明,本算法能在比较短的时间内达到比较好的压缩效果。

参　考　文　献

1　纪　震,周家锐,姜　来. DNA 序列数据压缩综述. 电子学报,2010;(5):1113—1121

2　张丽霞,张义青,林丕源. 基于字符和 0/1 码的 DNA 压缩模式匹配算法. 计算机应用研究,2007;24(9):22—24

3　Ferreira P J S G, Neves A J R, *et al*. Explorin three-base periodicity for DNA compression and modeling. Proceeding of the IEEE Conference on Acoustics ,Speech and Signal Processing. Toulouse ,2006: 877—880

4　Chen X, Kwong S, *et al*. A compression algorithm for DNA sequences and its applications in genome comparison. Procceeding of the 10th Workshop on Genome Informatics. Tokyo: GIW, 1999: 51—61

5　Korodi G, Tabus I, *et al*. DNA sequence compression-based on the normalized maximum likelihood model. IEEE Signal Processing Magazine,2007;24(1):47—53

6　纪　震,周家锐,朱泽轩. 基于生物信息学特征的 DNA 序列数据压缩算法. 电子学报,2011;(5):991—996

7　许忠能. 生物信息学. 北京. 清华大学出版社,2008

表 2　本文算法与经典算法压缩率对比

DNA 序列	长度	Biocompress	Gencompress	BioLZMA	本文算法
CHMPXX	121 024	15.76%	16.52%	20.54%	17.22%
CHNTXX	155 844	19.41%	19.31%	19.83%	18.75%
HEHCMVCG	229 354	7.60%	7.64%	15.41%	8.88%
HUMDYSTROP	33 770	3.69%	3.84%	1.69%	5.79%
HUMGHCSA	66 495	34.63%	45.14%	17.4%	40.27%
HUMHDABCD	58 864	6.15%	9.54%	8.16%	11.01%
HUMHBB	73 308	6.16%	9.29%	5.54%	10.56%
VACCG	191 737	11.93%	11.93%	23.92%	12.40%

(下转第 7514 页)

Adsorption Mechanism of Indigenous Microbial Activator in Porous Medium

XIE Kun[1], XIA Wen-jie[2], DONG Han-ping[1], YU Li[1], HUANG Li-xin[2]

(Institute of Porous Flow and Fluid Mechanics, Chinese Academy of Science[1], Langfang 065007, P. R. China;
Research Institute of Petroleum Exploration and Development-Langfang[2], Langfang 065007, P. R. China)

[**Abstract**] Nutrition activator plays the important role in the indigenous microbial enhanced oil recovery (IMEOR). In order to determine the absorption mechanism of the activator in the porous medium, the adsorption of glucose and nitrate on the quartz sand were investigate by the static and dynamic adsorption methods under the simulation condition. The static results showed that the adsorption of glucose and nitrate on the quartz sand were conformed to the Langmuir adsorption model, the maximum adsorption quantity of glucose and nitrate is 1.024 mg/g and 0.897 mg/g respectively. The dynamic results demonstrate that the retention of glucose and nitrate in porous medium are 0.113 3 mg/g and 0.067 4 mg/g respectively, and the dynamic retention is greater than the maximum static adsorption quantity.

[**Key words**] indigenous nicrobial oil recovery activator porous medium adsorption

(上接第 7509 页)

A New Compression Scheme for DNA Sequences Based on Statistical Analysis and Segmented Codebook

XIONG Wen-ping, SUN Ji-feng

(School of Electronic and Information Engineering, South China University of Technology, Guangzhou 510640, P. R. China)

[**Abstract**] DNA sequence is divided into short sequences with a length of 64 bases in every group. According to the different bases arrangement characteristics of each small paragraph, the specific nucleotides triplet is encoded which repeats the most times in a small paragraph with a particular codebook and a compression scheme for DNA data based on statistical analysis and segmented codebook is put fonward. Thus achieve the purpose of DNA data compression. The experiments show that the proposed algorithm can achieve a good performance in compressing most of the common benchmark sequences.

[**Key words**] DNA sequences statistical analysis codebook segmented encoding